

# COMPARING MEASUREMENT MODELS FOR TRACKING PEOPLE IN THERMAL IMAGES ON A MOBILE ROBOT

*André Treptow*

Department of Computer Science  
University of Tuebingen  
Tuebingen, Germany  
treptow@informatik.uni-tuebingen.de

*Grzegorz Cielniak and Tom Duckett*

AASS, Department of Technology  
University of Oerebro  
Oerebro, Sweden  
{grzegorz.cielniak, tom.duckett}@tech.oru.se

## ABSTRACT

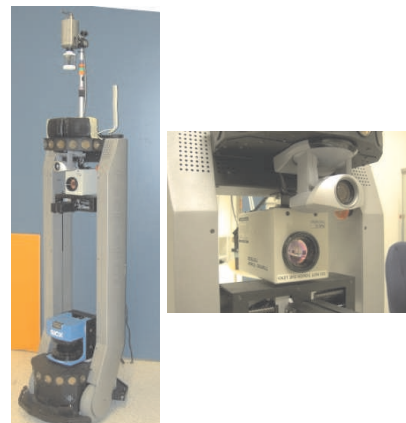
While most vision systems for tracking people on mobile robots use skin color information, we present an approach using thermal images and two different measurement models together with a Particle Filter. With this method a person can be detected independently from current light conditions and in situations where no skin color is visible (the person is not close or does not face the robot). The results show that a measurement model that was learned from local greyscale features improved on the performance of an elliptic contour model, and that both models could be used in combination to further improve performance with minimal extra computational cost.

## 1. INTRODUCTION

Vision-based detection, tracking and identification of humans on mobile robots is a challenging task. The ability to interact with people in populated environments is important for robots that fulfill tasks in cooperation with humans (e.g., service robots, inspection tasks, surveillance). Recently, systems for human-robot interaction that are able to locate the position of a person facing the robot have been developed. However, these approaches assume that people are close to the robot and face toward it so that methods based on skin color and face detection can be applied: Wilhelm et al. [11] track regions in the image which have skin color and combine this information with sonar data to get an estimate of the position of a person that is close to the robot. In a second step they use a face detector to get the position of the face in the image. Barreto et al. [6] describe a human-robot interface that relies purely on a face detector in combination with face recognition based on PCA. Similar work can be found in [7] where a detected face region is tracked with skin color information. Lang et al. [10] combine several cues including sonar, laser scanner, sound localisation and color image processing.

The work presented here is part of a robotic security

guard project, where one task for the mobile robot is to identify people in the building while patrolling. In this scenario the robot must be able to detect a person even from larger distances and it cannot be assumed that the person faces the direction of the robot. Therefore skin color cannot be used as a cue for the position of a person in the image. In this paper we address this problem and compare two methods to detect and track people in thermal images: A contour model which measures a persons shape probability and a model based on simple grey value features. Both models are integrated into a Particle Filter tracking algorithm. Our experimental platform is an ActivMedia PeopleBot mobile robot that is equipped with several sensors including a thermal camera and a pan-tilt camera unit (see figure 1). The paper is organised as follows: In section 2 we will introduce the whole system to identify people on the mobile security robot. In the following we concentrate on the part of tracking people in thermal images and describe the contour model (section 2.2), the grey value feature model (section 2.3) and the combination of both models (section 2.4). In section 3 the different models are evaluated and compared on a set of test sequences.



**Fig. 1.** ActivMedia Peoplebot, thermal and pan tilt camera.

## 2. METHOD

Our complete system to identify people in real time on a mobile security robot is shown in figure 2. The system can be divided into 4 parts. First of all, the robots starts in the search mode where it tries to detect a person based on the information from the thermal camera. If a person is detected in the thermal image the robots drives toward the person while tracking. This part is the attention system where the robots tries to get a rough estimate of the person’s position based on thermal images. If the robot is close to a person we use grey value images from the pan tilt camera to track the face. While tracking the face, images from the face tracker are fed into the recognition system to update an estimate of the identity of the person. In the following we concentrate on the first part of the system which is tracking in thermal images.

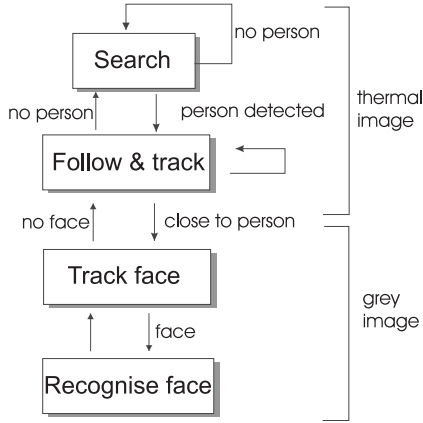


Fig. 2. Overview over the people identification system.

### 2.1. Tracking people in thermal images

The advantage of using sensor information from a thermal camera is that a person in the thermal image has a very distinctive profile so that the person can be clearly separated from the background. In figure 3 one can see that in the color image there is hardly any skin color visible if the person is further away, even though the person faces toward the camera. On the other hand one can easily detect the person in the same scene shown by the thermal image. However, apart from the work published in [3], where Cielniak and Duckett use image segmentation based on thresholding, noise filtering and morphological operations, there is hardly any published work on using thermal sensor information to detect humans on mobile robots until now. Infrared sensors have been applied to detect pedestrians in a driving assistance system: Bertozzi et al. [8] use a template based approach while Nanda and Davis [4] apply different image

filtering techniques. Meis et al. [12] also filter the whole image and classify based on the symmetry calculated for gradients. Xu et al. [2] employ a classification method based on a support vector machine. However, template based detection as well as SVM classification and image filtering over the whole image is time consuming. Xu et al. reported a frame-rate of their system of about 5Hz and the frame rate of system proposed in [4] lies between 3Hz and 11Hz depending on the image resolution.



Fig. 3. Person in color and thermal image.

To track a person in the thermal image we use a Particle Filter and two different measurement models which are very fast to calculate. Particle Filters [1], which are also known as Condensation [5] in the field of computer vision, have become quite popular in recent years for estimating the state of a system at a given time based on current and past measurements. The probability  $p(X_t|Z_t)$  of a system being in the state  $X_t$  given a history of measurements  $Z_t = \{z_0, \dots, z_t\}$  is approximated by a set of  $N$  weighted samples:

$$S_t = \{x_t^{(i)}, \pi_t^{(i)}\}, i = 1 \dots N. \quad (1)$$

Each  $x_t^{(i)}$  describes a possible state weighted with  $\pi_t^{(i)}$  which is proportional to the likelihood that the system is in this state. Particle Filtering consists of three main steps:

1. Create new sample set  $S_{t+1}$  by resampling from the old sample set  $S_t$  based on the sample weights  $\pi_t^{(i)}, i = 1 \dots N$
2. Predict sample states based on the dynamic model  $p(x_{t+1}^{(i)}|x_t^{(i)}), i = 1 \dots N$
3. Calculate new weights by application of the measurement model:  $\pi_{t+1}^{(i)} \propto p(z_{t+1}|X_{t+1} = x_{t+1}^{(i)}), i = 1 \dots N$ .

The estimate of the system state at time  $t$  is the weighted mean over all sample states:

$$\hat{X}_t = E(S_t) = \sum_{i=1}^N \pi_t^{(i)} x_t^{(i)}. \quad (2)$$

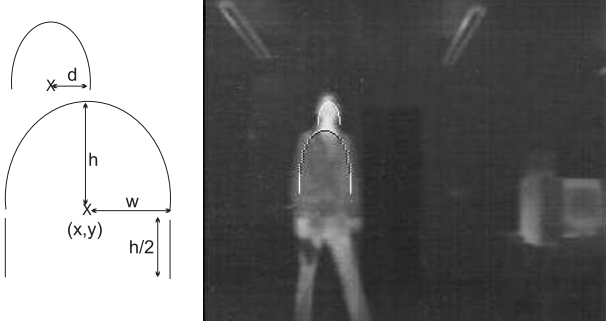


Fig. 4. The elliptic measurement model in thermal images.

## 2.2. Elliptic contour model

The contour measurement model to estimate the position of a person in the image consists of two elliptic measurements: one ellipse describes the position of the body part and one ellipse measures the position of the head part. Therefore, we end up with a 9-dimensional state vector:

$$x_t = (x, y, w, h, d, v_x, v_y, v_w, v_h) \quad (3)$$

where  $(x, y)$  is the mid-point of the body ellipse with a certain width  $w$  and height  $h$ . The height of the head is calculated by dividing  $h$  by a constant factor. The displacement of the middle of the head part from the middle of the body ellipse is described by  $d$ . We also model velocities of the body part as  $(v_x, v_y, v_w, v_h)$ . The elliptic contour model can be seen in figure 4. To calculate the weight  $\pi_t^{(i)}$  of a

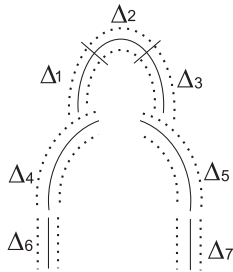


Fig. 5. Elliptic model divided into 7 sections.

sample  $i$  with state  $x_t^{(i)}$  we divide the ellipses into different regions (see figure 5) and for each region  $j$  the image gradient  $\Delta_j$  between pixels in the inner part and pixels in the outer part of the ellipse is calculated. The gradient is maximal if the ellipses fit to the contour of a person in the image data. The weight  $\pi_t^{(i)}$  is then calculated as the sum of all gradients multiplied with a weight factor:

$$\pi_t^{(i)} = W \cdot \sum_{j=1}^m \Delta_j \quad (4)$$

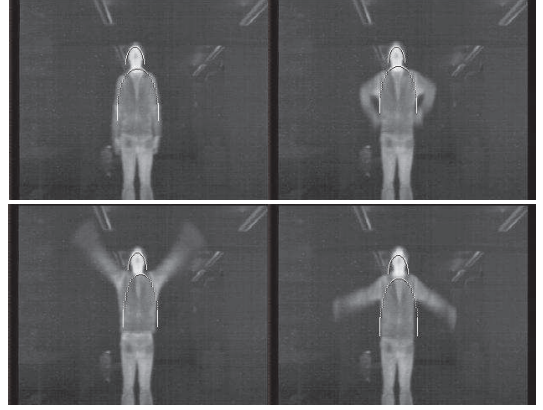


Fig. 6. Tracking with different arm positions.

with

$$W = \sum_{j=1}^m w_j, \quad w_j = \begin{cases} 0 & : \text{if } \Delta_j < t_h \\ \omega_j & : \text{otherwise} \end{cases} \quad (5)$$

The value  $t_h$  defines a gradient threshold and the weights  $\omega_j$  are chosen in a way that the shoulder parts have lower weight to minimize the measurement error that occurs due to different arm positions (see figure 6). The dynamic model that we use for the Particle Filter is a simple random walk: we model a movement with constant velocity plus small random changes. Our approach to track the contour of a person in the image is similar to the work by Isard and Blake [5] for tracking people in a grey image. However, they use a spline model of the head and shoulder contour which cannot be applied in our case because in situations where the person is far away or visible in a side view, there is no recognisable head-shoulder contour. The elliptic contour model is able to cope with these situations. The second advantage of using our contour model is that it can be calculated very quickly due to the fact that we measure only differences between pixel values on the inner and outer part of the ellipse. In figure 7 one can see the results of tracking a person under different views at different distances. Starting with a frontal view the person turns to a side view and back view.

## 2.3. Feature model

To build a measurement model based on grey value features, we use the algorithm proposed by Viola and Jones [9], which is considered to be one of the fastest systems to detect objects in grey value images. With this approach, classifiers that consist of simple features are learned offline on a given training set. Each so-called “strong classifier” is a linear combination of a number of “weak classifiers” which are simple threshold classifiers based on a single grey value feature. The features can be calculated very quickly on a



Fig. 7. Tracking under different views using the elliptic measurement model.

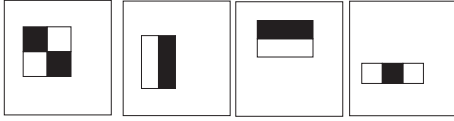


Fig. 8. Four different types of rectangle features: The sum of pixels in the white boxes are subtracted from the sum of pixels in the black areas.

so-called integral image: an integral image  $II$  over an image  $I$  is defined as  $II(x, y) = \sum_{x' \leq x, y' \leq y} I(x', y')$ . Good features that are able to discriminate between positive and negative object examples are selected with a boosting mechanism to build the final strong classifiers (for details see figure 9 and [9]). We train a single strong classifier and instead of scanning the classifier over the whole image at every location and every scale to detect a person we use Particle Filtering again: each sample describes a possible person located at position  $(x, y)$  and having the width  $w$ . Therefore, the state vector becomes  $x_t = (x, y, w)$ . The height  $h$  can be calculated by multiplying  $w$  with a constant factor depending on the size of the training images. To calculate the weight  $\pi_t^{(i)}$  the classifier is evaluated at the particle's position. Instead of using the binary output of the classifier, we rate each sample according to the weighted sum of all  $T$  features which are part of the strong classifier:  $\pi_t^{(i)} = \delta \sum_{j=1}^T \alpha_j h_j(x_t)$  where  $\alpha_j, h_j$  are the weighted weak classifiers (see [9]).

#### 2.4. Combination of contour and feature based model

To combine the two models we propose a cascaded model evaluation: For each sample, the feature model is evaluated first, due to the fact that this evaluation is less time consuming than the calculation of the contour. If the evaluation of the feature model returns a negative classification, the weight of the particle is set to zero. In case of a positive classification, the weight of the particle is set to the result of the contour evaluation.

1. Input: Training examples  $(x_i, y_i)$ ,  $i = 1..N$  with positive ( $y_i = 1$ ) and negative ( $y_i = 0$ ) examples.

2. Initialization: weights  $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$  with  $m$  negative and  $l$  positive examples

3. For  $t=1, \dots, T$ :

(a) Normalize all weights

(b) For each feature  $j$  train classifier  $h_j$  with error  $\epsilon_j = \sum_i w_{t,i} |h_j(x_i - y_i)|$

(c) Choose  $h_t$  with lowest error  $\epsilon_t$

(d) Update weights:  $w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$  with

$$e_i = \begin{cases} 0 & : x_i \text{ correctly classified} \\ 1 & : \text{otherwise} \end{cases}$$

and  $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$

4. Final strong classifier:

$$h(x) = \begin{cases} 1 & : \sum_{t=1}^T \alpha_t h_t(x) \geq 0.5 \sum_{t=1}^T \alpha_t \\ 0 & : \text{otherwise} \end{cases}$$

with  $\alpha_t = \log(\frac{1}{\beta_t})$

Fig. 9. Adaboost learning algorithm as proposed in [9].

### 3. EXPERIMENTS

#### 3.1. Test sequences

We recorded test sequences of thermal images with 17 different persons. In each sequence a person stood 4 to 5 me-



**Fig. 10.** Comparing elliptic measurement (left), feature model (middle) and combination (right).

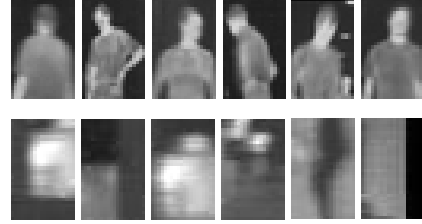
ters from the robot in an unconstrained indoor environment, and the robot was started facing away from the person so that it had to turn around and search for the person in the thermal image. While tracking the person in the thermal image, the robot approached and stopped in front of the person. After that, the person walked to a different position behind the robot and the robot approached a second time, so that we recorded each person under two different conditions. The length of the recorded sequences varied from 700 to 1400 frames per person with an image resolution of  $320 \times 240$ . One sequence had been used to get negative example images to train the feature model (see section 3.2) so that the remaining 16 sequences could be used for testing. To get ground truth information about the position of a person in the thermal images we used a semi-automatic method to segment the sequences: the result from a segmentation based on a flood-fill algorithm was corrected by hand to extract the exact region in the thermal image containing a person.

### 3.2. Training

To train the feature based model, we collected a database with 106 images (positive examples) showing the upper body part of different people under different views and 5784 images that do not show a person (negative examples) in the thermal image. The images have the size of  $20 \times 32$  pixels and are normalised to have zero mean and a standard deviation. Figure 11 shows some of the images from the training set. The model had been trained until the whole database had been classified correctly which resulted in a strong classifier with one cascade level consisting of 29 different features.

### 3.3. Comparison

To compare the different measurement models we evaluated the tracking performance on the test sequences. In the Particle Filter we used a total of 300 particles, and the weighted mean of the best 20% of all particles of the tracker was compared to the ground truth data for all test sequences. If a



**Fig. 11.** Images from the training set: person (top row) and none-person (bottom row) images.

person was detected in a frame and the person was visible in the ground truth segmentation, we calculated a detection accuracy  $d_{acc}$  as follows:

$$d_{acc} = \frac{2 \cdot n_{overlap}}{n_{detected} + n_{real}}, \quad (6)$$

where  $n_{overlap}$  is the number of overlapping pixels between the box around the true person position and the detected position.  $n_{detected}$  is the number of pixels in the box, which the tracker returns and  $n_{real}$  is the number of pixels in the rectangle around the true person position. Based on the detection accuracy we calculated the following values on each test sequence to evaluate the performance of the tracker:

- False positive rate  $FPR = \frac{N_F}{N_N}$  with  $N_F$ =number of frames where a person was detected but not visible in ground truth,  $N_N$ =total number of frames where no person was visible.
- Detection rate  $DR = \frac{N_D}{N_P}$  with  $N_D$ = number of frames where person was detected (with  $d_{acc} \geq 0.6$ ) and visible in ground truth,  $N_P$ =total number of frames where a person was visible.
- Classification rate  $CR$ : percentage of all frames which were correctly classified ( $d_{acc} \geq 0.6$  or no person in ground truth and no person detected).

Table 1 shows the results of the evaluation. As one can see, the feature based model performs slightly better than

Method	$d_{acc}$	CR	DR	FPR	time
Contour	79.1	88.9	86.1	7.4	11ms
Grey features	88.6	91.9	96.0	12.7	11ms
Combination	92.2	92.2	90.0	4.8	14ms

**Table 1.** Tracking results on test sequences.

the elliptic model. The reason for this is that in some cases the contour model does not fit exactly to the body part of the person which results in a bad estimate of the whole body (see figure 10). However, the main disadvantage of using the feature model is that we just get a rough estimate about the position of the upper body part of the person while the contour model gives us the position of the head relative to the body. This information could be necessary for e.g. a face tracking system on a higher level. Therefore, we propose the usage of the combination of both models to achieve the most accurate tracking results. As shown in table 1, with the combination of both models we are able to get the highest detection accuracy with the lowest number of false detections. On an Athlon XP1600 processor, the time for processing one frame with the contour model is 11ms while the feature model requires 7ms for classifier evaluation and 4ms to calculate the integral image. The combination of both models requires 14ms which is equivalent to a frame-rate of 71 Hz and leaves enough computational resources for other high-level tasks such as planning, navigation, face recognition etc.

#### 4. CONCLUSION AND FUTURE WORK

In this paper we compared two different measurement models to detect and track people in thermal images using particle filtering. To get best tracking results, we proposed the combination of a contour based model with an offline learned grey value feature model. This method is robust and can be evaluated in real time. Until now, the tracker will always lock onto a single person (the person that has highest measurement probability in the thermal image) but we are currently extending our approach to multiple persons using multiple clusters of particles.

#### 5. REFERENCES

- [1] N. de Freitas A. Doucet and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer, New York, 2001.
- [2] F. Xu, X. Liu and K. Fujimura. Pedestrian Detection and Tracking with Night Vision. *IEEE Transactions on Intelligent Transportation System*, 5(4), 2004.
- [3] G. Cielniak and T. Duckett. Person Identification by Mobile Robots in Indoor Environments. In *Proc. IEEE Int. Workshop on Robotic Sensing (ROSE 2003)*, Örebro, Sweden, 2003.
- [4] H. Nanda and L. Davis. Probabilistic Template based Pedestrian Detection in Infrared Videos. In *IEEE Intelligent Vehicle Symposium*, Versailles, France, 2002.
- [5] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.
- [6] J. Barreto, P. Menezes and J. Dias. Human-Robot Interaction based on Haar-like Features and Eigenfaces. In *Proc. of the 2004 IEEE International Conference on Robotic and Automation (ICRA 04)*, pages 1888–1893, New Orleans, LA, 2004.
- [7] L. Brèthes, P. Menezes, F. Lerasle and J. Hayet. Face Tracking and Hand Gesture Recognition for Human-Robot Interaction. In *Proc. of the 2004 IEEE International Conference on Robotic and Automation (ICRA 04)*, pages 1901–1906, New Orleans, LA, 2004.
- [8] M. Bertozzi, A. Broggi, P. Grisleri, T. Graf and M. Meinecke. Pedestrian Detection in Infrared Images. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 662–667, Columbus, USA, 2003.
- [9] P. Viola and M. J. Jones. Robust real-time object detection. In *Proc. of IEEE Workshop on Statistical and Theories of Computer Vision*, 2001.
- [10] S. Lang, M. Kleinhenrich, S. Hohener, J. Fritsch, G. A. Fink and G. Sagerer. Providing the Basis for Human-Robot-Interaction: A Multi-Modal Attention System for a Mobile Robot. In *Proc. Int. Conf. on Multimodal Interfaces*, pages 28–35, Vancouver, Canada, 2003.
- [11] T. Wilhelm, H.-J. Böhme and H.-M. Gross. A Multi-Modal System for Tracking and Analyzing Faces on a Mobile Robot. *Robotics and Autonomous Systems*, 48(1):31–40, 2004.
- [12] U. Meis, W. Ritter and H. Neumann. Detection and classification of obstacles in night vision traffic scenes based on infrared image. In *Proc. IEEE Intelligent Transportation Systems*, pages 1140–1144, Shanghai, China, 2003.